Муниципальное бюджетное образовательное учреждение “Лицей №40”
Центр развития компетенций “Мультилаб” ЭН+


Алексей Зотов


Дмитрий Пудалов


Тимофей Бацын


Дмитрий Белокрылов

КОМАНДА

FIRSTEAM

ПРЕДСТАВЛЯЕТ ИННОВАЦИОННЫЙ ПРОЕКТ:

Автономная система обнаружения и обезвреживания БПЛА

актуальная проблема

поскольку инфраструктура всё чаще подвергаются атакам с применением БПЛА, мы решили остановиться на проблеме, связанной с защитой важных объектов

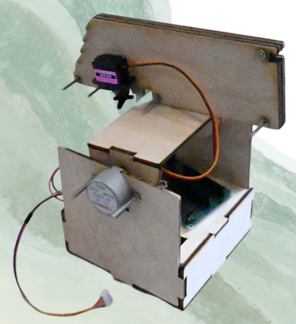
ВАРИАНТЫ РЕШЕНИЯ

Заглушение связи БПЛА с оператором по средствам радио-электронной борьбы

Обнаружение посредством КЗ с последующим противодействием

Запуск само-управляемого дрона, ориентирующегося на источник звука

РОБОТИЗИРОВАННЫЕ РЕШЕНИЯ



1 вариант

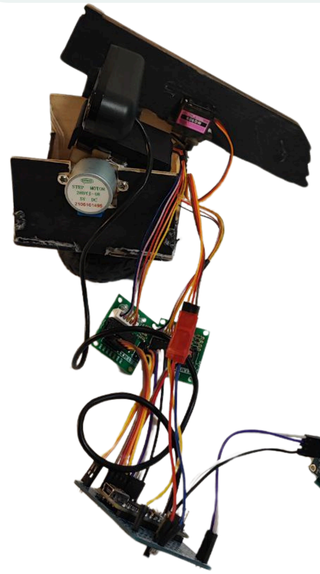
Мы использовали в качестве условного дрона деталь определённого цвета, которого нет на фоне. Камера была статичной, вращался только исполнительный блок. Метод обнаружения: поиск скопления пикселей определенного цвета и наведение на геометрический центр скопления

Минусы:
- сильно зависит от освещения;
- малый угол обзора;
- работает не на любом фоне;
- не всегда правильно наводится;
- ненадёжная конструкция

2 вариант

Мы усилили корпус, развели плату на макетной основе, увеличили боезапас относительно первой версии. Установили камеру на исполнительный блок, что сильно увеличило угол обнаружения и точность наведения

Минусы:
- всё ещё сильно зависит от освещения;
- работает не на любом фоне;
- не всегда правильно наводится;
- работает только на условной цели;
- атака происходит вне зависимости от расстояния до условного БПЛА

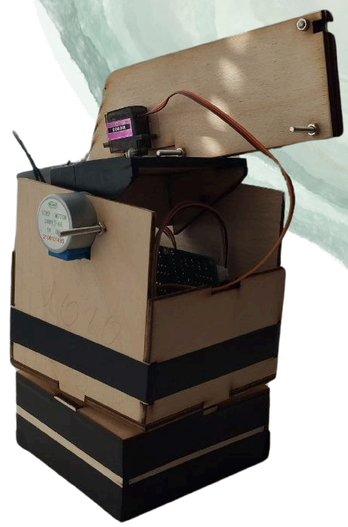


ОБУЧЕНИЕ НЕЙРОСЕТИ

1 этап
- фотографируем дрон с разных ракурсов, в разных помещениях с разным освещением;
- размечаем на классы (drone/air);
- создаём .txt-файлы с коорд. выделенных областей объектов

2 этап
- делим на пакеты train (обучение) и test (тестирование);
- подключение docker-контейнера для развертывания машинного обучения;
- запуск машинного обучения

3 этап
- проверка результатов обучения на пакете test;
- дополнительное обучение нейросети;
- интеграция результатов обучения в основной python-код



консультанты проекта:

Грачёв Алексей Анатольевич - arduino;
Глушков Дмитрий Вадимович - нейросети;
Гринберг Евгений Евгениевич - нейросети

финальный вариант
- обнаруживает реальный дрон, а не условный объект;
- использует машинное обучение;
- способен отличит БПЛА от иных объектов;
- может поддерживать разные классы объектов, находящихся в зоне видимости;
- не зависит от освещения;
- работает на любом фоне;
- гораздо лучше наводится;
- атака зависит от расстояния до БПЛА

ПРОГРАММНАЯ ЧАСТЬ
- от нейросети мы получаем координаты дрона;
- в python данные координаты преобразуются в количество шагов, необходимое мотору для наведения по осям;
- отсылаем команды наведения на arduino;
- робот следит за дроном с использованием ПИД-регулятора
- при определенном расстоянии до дрона робот производит залп

АППАРАТНАЯ ЧАСТЬ
- Arduino Nano управляет исполнительным механизмом;
- 2 шаговых мотора отвечают за наведение;
- сервомотор обеспечивает стрельбу
- датчик расстояния определяет расстояние до предполагаемой цели;
- веб-камера передаёт изображение в ЦВС;
- центральная вычислительная система в виде ноутбука